

PATENT ABSTRACTS OF JAPAN

(11)Publication number : 08-286920

(43)Date of publication of application : 01.11.1996

(51)Int.Cl.

G06F 9/44

G06F 9/44

G06F 15/16

(21)Application number : 07-087861

(71)Applicant : FUJITSU LTD

(22)Date of filing : 13.04.1995

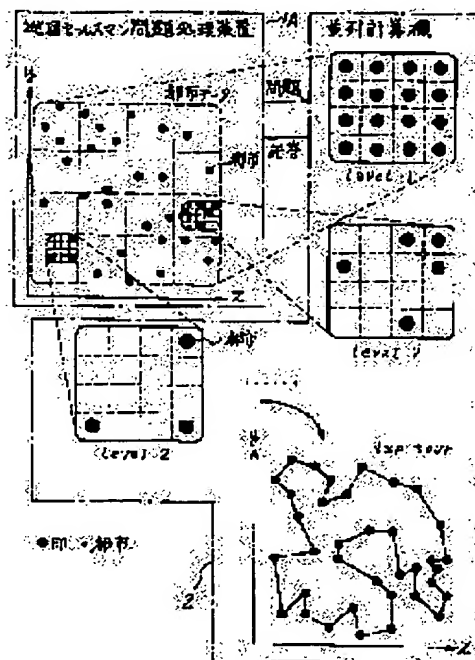
(72)Inventor : MORIOKA KAYOKO
ISHII MITSUO
KAWAMURA KAORU
INADA YOSHIE
MITO HIDEKI

(54) TRAVELING SALESMAN PROBLEM PROCESSOR

(57)Abstract:

PURPOSE: To speedily find a result close to an optimum solution by processing a large scale traveling salesman problem at high speed by finding the solution of the traveling salesman problem at high speed.

CONSTITUTION: A traveling salesman problem processor 1A is provided with an entire problem processing means for finding the solution by processing the entire problem while using a parallel processor 2 by preparing an entire problem by dividing a minimum rectangle including an entire city corresponding to the number of cells of the parallel computer 2 based on an applied city coordinate and allocated the divided city to the respective cells, and a partial problem processing means for finding the solution by processing a partial problem while using the parallel computer 2 by preparing that partial problem by further hierarchically dividing the minimum rectangle including the city groups of non-found routes corresponding to the number of cells of the parallel computer 2 and allocating these hierarchical city data to the respective cells when the route of the solution found by the parallel computer 2 does not cover the whole city.



LEGAL STATUS

[Date of request for examination]

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number]

[Date of registration]

[Number of appeal against examiner's decision
of rejection]

[Date of requesting appeal against examiner's
decision of rejection]

[Date of extinction of right]

Copyright (C); 1998,2000 Japan Patent Office

(19)日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11)特許出願公開番号

特開平8-286920 ✓

(43)公開日 平成8年(1996)11月1日

(51)Int.Cl. ⁶	識別記号	庁内整理番号	F I	技術表示箇所
G 0 6 F 9/44	5 5 2	7737-5B	G 0 6 F 9/44	5 5 2
	5 5 0	7737-5B		5 5 0 Q
15/16	3 8 0		15/16	3 8 0 Z

審査請求 未請求 請求項の数6 O L (全 13 頁)

(21)出願番号 特願平7-87861

(22)出願日 平成7年(1995)4月13日

(71)出願人 000005223

富士通株式会社

神奈川県川崎市中原区上小田中4丁目1番1号

(72)発明者 森岡 佳代子

神奈川県川崎市中原区上小田中1015番地
富士通株式会社内

(72)発明者 石井 光雄

神奈川県川崎市中原区上小田中1015番地
富士通株式会社内

(74)代理人 弁理士 今村 辰夫 (外1名)

最終頁に続く

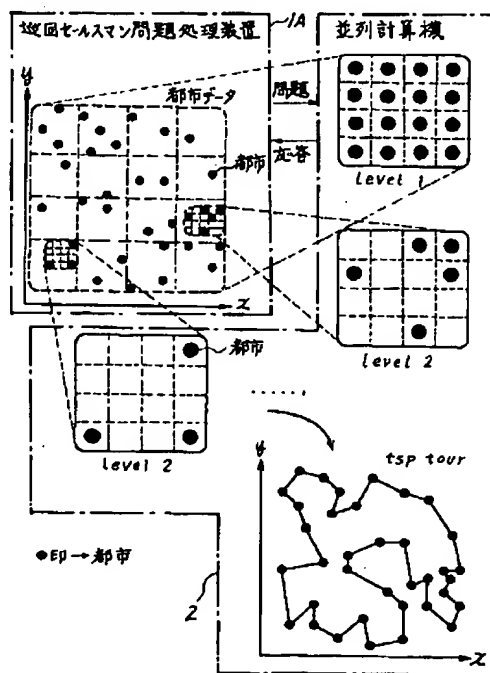
(54)【発明の名称】 巡回セールスマン問題処理装置

(57)【要約】

【目的】 本発明は巡回セールスマン問題処理装置に関し、巡回セールスマン問題の解を高速に求め、大規模巡回セールスマン問題を高速に処理し、最適解に近いものをできるだけ早く求められるようにすることを目的とする。

【構成】 巡回セールスマン問題処理装置1Aは、与えられた都市座標を基に全都市を含む最小矩形を並列計算機2のセル数に応じて分割し、分割した都市を各セルに割り当てることで全体問題を作成し、全体問題を並列計算機2に処理させて解を求める全体問題処理手段と、並列計算機2で求められた解の経路が全都市を巡るものではない場合、求められていない経路の都市群を含む最小矩形を更に並列計算機2のセル数に応じて分割することで階層化し、この分割して階層化した都市データを各セルに割り当てることで部分問題を作成し、その部分問題を並列計算機に処理させて解を求める部分問題処理手段を備えた。

本発明の原理説明図



1

【特許請求の範囲】

【請求項 1】 メッシュ状に接続した複数のセルと、各セル間を接続した通信ネットワークを有する並列計算機を使用して巡回セールスマン問題の解を求める巡回セールスマン問題処理装置において、

与えられた都市座標を基に全都市を含む最小矩形を並列計算機のセル数に応じて分割し、この分割した都市データを各セルに割り当てることで全体問題を作成し、その全体問題を並列計算機に処理させて解を求める全体問題処理手段と、

並列計算機で求められた解の経路が全都市を巡るものではない場合、求められていない経路の都市群を含む最小矩形を、更に並列計算機のセル数に応じて分割することで階層化し、この分割して階層化した都市データを各セルに割り当てることで部分問題を作成し、その部分問題を並列計算機に処理させて解を求める部分問題処理手段を備えていることを特徴とした巡回セールスマン問題処理装置。

【請求項 2】 前記部分問題処理手段は、部分問題を作成する直前までに求められている巡回経路を基に、部分問題における代表都市の直前の訪問都市の位置が部分問題の中のどの部分に割り当てられているかを調べ、そこから距離が一番近いセルに割り当てられている都市を部分問題の出発都市に設定する出発都市設定手段を備えていることを特徴とした請求項 1 記載の巡回セールスマン問題処理装置。

【請求項 3】 前記部分問題処理手段は、部分問題を作成する直前までに求められている巡回経路を基に、部分問題における代表都市の次の訪問都市の位置が部分問題の中のどの部分に割り当てられているかを調べ、そこから距離が一番近いセルに割り当てられている都市を到着都市に設定する到着都市設定手段を備えていることを特徴とした請求項 1 記載の巡回セールスマン問題処理装置。

【請求項 4】 前記部分問題処理手段は、部分問題を作成する際、出発都市、或いは到着都市に対応する位置に都市がない場合は、その位置に疑似的に作成した疑似都市を設定する疑似都市設定手段を備えていることを特徴とした請求項 1 記載の巡回セールスマン問題処理装置。

【請求項 5】 前記並列計算機から部分問題の解が得られた場合、その経路中に疑似都市が含まれているか否かを判断し、疑似都市が含まれていた場合は、その疑似都市を除いた経路を求めて、その経路を全体の経路に組み込むと共に、疑似都市が含まれていない場合は、そのままの経路を全体の経路に組み込む経路組み込み処理手段を備えていることを特徴とした請求項 1 記載の巡回セールスマン問題処理装置。

【請求項 6】 前記並列計算機から部分問題の解が得られた場合、その経路中に重複した経路が存在するか否かを判断し、重複した経路が存在した場合、到着都市の位

2

置をずらして経路を修正する経路重複修正手段を備えていることを特徴とした請求項 1 記載の巡回セールスマン問題処理装置。

【発明の詳細な説明】

【0001】

【産業上の利用分野】 本発明は、メッシュ状に接続した複数のプロセッサエレメント（以下「セル」と記す）と、それらの間を接続した通信ネットワークを有する並列計算機を使用して巡回セールスマン問題を高速に解き、かつ最適解を求めることが可能な巡回セールスマン問題処理装置に関する。

【0002】 巡回セールスマン問題は、2次元平面に散在する都市を一巡する経路の内、最短経路を探索する問題である。例えば、複数の仕事があって、それらの仕事には、それぞれ次の仕事にとりかかるまでに準備時間が必要だとする。この時、それぞれの仕事間の準備時間を巡回セールスマン問題における都市間の距離と思うと、全ての仕事を最短時間で終わらせる仕事の組み合わせは、巡回セールスマン問題の最適解と同じである。

【0003】 前記巡回セールスマン問題においては、一巡する経路の数は、 N 個の都市に対して $(N-1)!$ だけ存在する。そして、都市の増加と共に都市の組み合わせによる経路の計算量は爆発的に増加するため、巡回セールスマン問題を階層化して並列計算機に処理させることにより、都市の組み合わせ数を抑え、処理の高速化を図る必要がある。

【0004】

【従来の技術】 従来、計算機を使用して巡回セールスマン問題（以下「TSP」と記す）を解くことが試行されてきた。この処理は、計算機に TSP を解くための都市情報等を入力し、2次元平面上に散在する全ての都市を一巡する経路の内、最も短い経路が探索できたら経路探索処理を終了するというものである。

【0005】 そして、前記 TSP の解法については種々のアルゴリズムが試されてきたが、都市数の増加と共に、都市の組み合わせは階乗のオーダーで増加して行く。このため、計算量が莫大なものになってしまうので、最適解を求めることは殆ど不可能であった。

【0006】 ところで、TSP は、典型的な組み合わせ問題として多くの研究者が取り組んできた問題である。最近では、ホップフィールド型ニューラルネットワークで解く方法が盛んに研究されている。また、階層化してその解を求める方法も研究されている。

【0007】 しかし、ホップフィールド型ニューラルネットワークでは、局所最小値に落ち込んで最適解に辿りつかないことがある。その他、従来からある階層化した解法においては、階層化することで解の精度が悪くなっていた。また、階層化することで処理に時間がかかっていた。

【0008】

3

【発明が解決しようとする課題】前記のような従来のものにおいては、次のような課題があった。

(1) : 前記TSPの解法については種々のアルゴリズムが試されてきたが、都市数の増加と共に、都市の組み合わせは階乗のオーダーで増加して行く。このため、計算量が莫大なものになってしまうため、最適解を求めることは殆ど不可能であった。

【0009】(2) : 例えば、ホップフィールド型ニューラルネットワークを利用してTSPを解く方法では、局所最小値に落ち込んで最適解に辿りつかないことがある。

(3) : その他、従来からある階層化した解法においては、階層化することで解の精度が悪くなることと、時間がかかるという課題がある。

【0010】本発明は、このような従来の課題を解決し、メッシュ構造の並列計算機を使用して、巡回セールスマン問題の解を高速に求められるようにすることを目的とする。

【0011】また、本発明は、巡回セールスマン問題をメッシュ構造の並列計算機のセル数に応じて分割し、階層化して処理することにより、大規模の巡回セールスマン問題を高速に処理し、最適解に近いものをできるだけ早く求めることができるようにすることを目的とする。

【0012】

【課題を解決するための手段】図1は本発明の原理説明図である。本発明は前記の課題を解決するため、巡回セールスマン問題処理装置を次のように構成した。

【0013】(1) : メッシュ状に接続した複数のセルと、各セル間を接続した通信ネットワークを有する並列計算機2を使用して巡回セールスマン問題の解を求める巡回セールスマン問題処理装置1Aにおいて、与えられた都市座標を基に全都市を含む最小矩形を並列計算機2のセル数に応じて分割し、この分割した都市データを各セルに割り当てることで全体問題を作成し、その全体問題を並列計算機2に処理させて解を求める全体問題処理手段と、並列計算機2で求められた解の経路が全都市を巡るものでない場合、求められていない経路の都市群を含む最小矩形を、更に並列計算機2のセル数に応じて分割することで階層化し、この分割して階層化した都市データを各セルに割り当てることで部分問題を作成し、その部分問題を並列計算機2に処理させて解を求める部分問題処理手段を備えている。

【0014】(2) : 前記(1)の巡回セールスマン問題処理装置において、部分問題処理手段は、部分問題を作成する直前までに求められている巡回経路を基に、部分問題における代表都市の直前の訪問都市の位置が部分問題の中のどの部分に割り当てられているかを調べ、そこから距離が一番近いセルに割り当てられている都市を部分問題の出発都市に設定する出発都市設定手段を備えている。

4

【0015】(3) : 前記(1)の巡回セールスマン問題処理装置において、部分問題処理手段は、部分問題を作成する直前までに求められている巡回経路を基に、部分問題における代表都市の次の訪問都市の位置が部分問題の中のどの部分に割り当てられているかを調べ、そこから距離が一番近いセルに割り当てられている都市を到着都市に設定する到着都市設定手段を備えている。

【0016】(4) : 前記(1)の巡回セールスマン問題処理装置において、部分問題処理手段は、部分問題を作成する際、出発都市、或いは到着都市に対応する位置に都市がない場合は、その位置に疑似的に作成した疑似都市を設定する疑似都市設定手段を備えている。

【0017】(5) : 前記(1)の巡回セールスマン問題処理装置において、並列計算機2から部分問題の解が得られた場合、その経路中に疑似都市が含まれているか否かを判断し、疑似都市が含まれていた場合は、その疑似都市を除いた経路を求めて、その経路を全体の経路に組み込むと共に、疑似都市が含まれていない場合は、そのままの経路を全体の経路に組み込む経路組み込み処理手段を備えている。

【0018】(6) : 前記(1)の巡回セールスマン問題処理装置において、並列計算機2から部分問題の解が得られた場合、その経路中に重複した経路が存在するか否かを判断し、重複した経路が存在した場合、到着都市の位置をずらして経路を修正する経路重複修正手段を備えている。

【0019】

【作用】前記構成に基づく本発明の作用を、図1に基づいて説明する。巡回セールスマン問題を解く場合、並列計算機2のセルに全都市を重複することなく割り当てることができれば、各セル間の通信で同期をとることにより、階層化することなく最適解を得ることができる。しかし、並列計算機2のセルの数が全都市数より少ない場合は、以下の手順で階層化して最適解に近いものを求めることができる。

【0020】先ず、巡回セールスマン問題処理装置1Aは、与えられた都市座標から全都市を含む最小矩形を $m \times n$ に分割して、並列計算機2の $m \times n$ のセルに巡回セールスマン問題の全都市をマッピングすることにより、全体問題(レベル1)を作成し、並列計算機2へ送信して処理を依頼する。この場合、全体問題は、前記 $m \times n$ のセルにマッピングした全都市を巡回する経路の内、最短経路を求める問題である。

【0021】前記全体問題を受信した並列計算機2は、 $m \times n$ のセルを用いて都市を巡回する経路の内、最短経路を求める処理を行う。そして、与えられた全ての都市を巡回する経路の内から最短経路が求められたら、その最短経路の経路データを巡回セールスマン問題処理装置1Aへ応答する。

【0022】前記応答を受信した巡回セールスマン問題

処理装置 1 A は、並列計算機 2 で求められた経路が全ての都市を巡回したものの可否かを判断する。その結果、求められた経路が全ての都市を巡回するものでないと判断した場合、新たに部分問題（レベル 2）を作成し並列計算機 2 へ部分問題の解を求める処理を依頼する。

【0023】前記部分問題を受信した並列計算機 2 は、部分問題に対し都市を巡回する経路の内、最短経路を求める処理を行う。そして、与えられた全都市を巡回する経路の内から最短経路が求められたら、その最短経路の経路データを巡回セールスマン問題処理装置 1 A へ応答する。

【0024】前記応答を受信した巡回セールスマン問題処理装置 1 A は、並列計算機 2 で求められた経路が全ての都市を巡回したものの可否かを判断する。その結果、求められた経路が全ての都市を巡回するものでないと判断した場合、更に階層化した部分問題を作成し並列計算機 2 へ部分問題の解を求める処理を依頼する。このようにして、全都市の経路が求められるまで、階層化した部分問題の処理を行う。

【0025】この場合、巡回セールスマン問題処理装置 1 A は、部分問題を作成する直前までに求められている巡回経路を基に、部分問題における代表都市の直前の訪問都市の位置が、部分問題の中のどの部分に割り当てられているかを調べ、そこから距離が一番近いセルに割り当てられている都市を部分問題の出発都市に設定する。

【0026】また、部分問題における代表都市の次の訪問都市の位置が部分問題の中のどの部分に割り当てられているかを調べ、そこから距離が一番近いセルに割り当てられている都市を到着都市に設定する。更に、部分問題を作成する際、出発都市、或いは到着都市に対応する位置に都市がない場合は、その位置に疑似的に作成した疑似都市を設定する。

【0027】前記のようにして各部分問題の解を得た巡回セールスマン問題処理装置 1 A は、その経路中に疑似都市が含まれているかを判断し、疑似都市が含まれていた場合は、その疑似都市を除いた経路を求めて、その経路を全体の経路に組み込むと共に、疑似都市が含まれていない場合は、そのままの経路を全体の経路に組み込む処理を行う。

【0028】この場合、並列計算機 2 から部分問題の解が得られたら、その経路中に重複した経路が存在するかの可否かを判断し、重複した経路が存在した場合、到着都市の位置をずらして経路を修正する処理を行う。

【0029】以上のようにメッシュ構造の並列計算機を使用して、巡回セールスマン問題の解を高速に求めることが可能になる。また、巡回セールスマン問題をメッシュ構造の並列計算機のセル数に応じて分割し、階層化して処理することにより、大規模の巡回セールスマン問題を高速に処理し、最適解に近いものをできるだけ早く求めることができる。

【0030】

【実施例】以下、本発明の実施例を図面に基づいて説明する。図 2～図 11 は本発明の実施例を示した図である。図 2～図 11 において、1 はホストコンピュータ、2 は並列計算機、3 は TSP 処理部（巡回セールスマン問題処理部）、4 は記憶装置、5 はセル（プロセッサエレメント）、6 は通信ネットワークを示す。なお、図 1 に示した巡回セールスマン問題処理装置 1 A は、前記ホストコンピュータ 1 に対応する。

10 【0031】§ 1：実施例のシステム構成の説明・・・図 2 参照

図 2 は実施例のシステム構成図である。実施例では、ホストコンピュータ 1 と並列計算機 2 からなるシステムにより巡回セールスマン問題（以下「TSP」と記す）の解を求める処理を行う。

【0032】前記システムにおいて、ホストコンピュータ 1 には、TSP の各種処理を行う TSP 処理部 3 と、TSP 処理に必要な都市データ等の各種情報を格納するための記憶装置 4 等が設けられている。この場合、記憶装置 4 には予め全都市の座標データが格納されている。

【0033】前記並列計算機 2 は、ホストコンピュータ 1 から依頼された TSP の問題データ（全体問題、及び部分問題）の処理を実行するものであり、例えば、メッシュ状に接続された複数のセル（プロセッサエレメント）5 と、これらの各セル 5 間を接続する通信ネットワーク 6 を備えている。

【0034】前記システムでは、ホストコンピュータ 1 の TSP 処理部 3 が TSP の問題データを作成し、この問題データをメッシュ構造の並列計算機 2 へ転送して TSP 処理を依頼する。そして、並列計算機 2 が解析して解が求まったら、その解のデータ（与えられた全ての都市間を巡回する最短経路のデータ）をホストコンピュータ 1 へ転送して応答する。TSP 処理部 3 は、前記並列計算機 2 からの応答を基に、TSP の全都市を巡る最短経路を求めるものである。

【0035】§ 2：実施例の処理の説明・・・図 3 参照
図 3 は実施例の処理フローチャートである。以下、図 3 に基づいて実施例の処理（全体の処理）を説明する。なお、S1～S8 は各処理ステップを示す。

40 【0036】TSP の解を求める場合、並列計算機 2 のセル 5 に全都市を重複することなく割り当てることができれば、各セル 5 間の通信で同期をとることにより、階層化することなく TSP の最適解を得ることができる。しかし、並列計算機 2 のセル 5 の数が全都市数より少ない場合は、以下の手順で TSP を階層化して最適解に近いものを求めることができる。

50 【0037】まず、TSP 処理部 3 は、記憶装置 4 から全都市の座標等の必要なデータを読み込み内部のメモリに格納して（S1）処理を開始する。そして、与えられた都市座標から全都市を含む最小矩形を $m \times n$ に分割し

て (m, n : 任意の整数)、並列計算機2の $m \times n$ のセルにTSPの全都市をマッピングする (S2)。

【0038】次にTSP処理部3は、前記マッピングした情報から最初に解くべき全体問題を作成し (S3)、前記作成した問題を並列計算機2へ送信して (S4) TSP処理 (与えられた全ての都市を巡回する経路の内、最短経路を求める処理) を依頼する。この場合、最初に解くべき全体問題は、前記 $m \times n$ のセルにマッピングした全都市 (全都市を $m \times n$ の都市として扱う) を巡回する経路の内、最短経路を求める問題である。

【0039】前記問題を受信した並列計算機2は、最初は $m \times n$ のセル5を用いて全ての都市を巡回する経路の内、最短経路を求める処理を行う。この場合、1つのセル5に複数の都市を含む都市群が存在することもあるが、1つのセルに割り当てられた都市群を1つの都市 (代表都市) と考えて最短経路を求める処理を行う。

【0040】そして、与えられた全都市を巡回する経路の内から最短経路が求められたら、その最短経路のデータをホストコンピュータ1のTSP処理部3へ転送して、前記依頼に対し応答する。

【0041】TSP処理部3は、並列計算機2からの応答を受信すると (S5)、その経路データを記憶装置4に格納しておく。そして、TSP処理部3は、並列計算機2で求められた前記経路が全ての都市を巡回したものか否かを判断する (S6)。その結果、求められた経路が全ての都市を巡回するものでないと判断した場合、新たに部分問題を作成し (S8)、前記S4の処理から繰り返して行う。

【0042】すなわち、全都市の経路が求められるまで、求められていない部分の都市群を含む最小矩形を $j \times k$ に再分割 (j, k : 任意の整数) して階層化する操作を繰り返して行う (レベル1→レベル2→レベル3・・・・)。このようにして、全都市を経由した経路が求まったら、経路探索結果のデータを出力 (表示、印刷等による出力) し (S7) 処理終了となる。

【0043】前記の処理において、並列計算機2から各部分問題の解 (求められた最短経路データ) を得たTSP処理部3は、求められた経路を全体の経路に組み込む処理を行い、記憶装置4に格納する。

【0044】以上のようにメッシュ構造の並列計算機を使用し、巡回セールスマン問題の解を高速に求めることが可能になる。また、巡回セールスマン問題をメッシュ構造の並列計算機のセル数に応じて分割し、階層化して処理することにより、大規模の巡回セールスマン問題を高速に処理し、最適解に近いものをできるだけ早く求めることができる。

【0045】§3：都市データの説明・・・図4参照
図4は都市データの説明図である。TSP処理部3は、前記のようにしてTSP処理 (TSPの解を求める処理) を行うが、この場合、記憶装置4内に図示のような

都市データを格納することで処理を行う。

【0046】すなわち、TSP処理部3がアクセスする記憶装置4には、各都市毎に、都市ID (都市の識別コード)、都市の実際の座標 ($x-y$ 座標)、都市の並列計算機における座標 (セルの位置座標)、インバース (inverse : 直前の訪問都市)、ネクスト (next : 次の訪問都市)、グループ (group : 同じグループの都市)、同じグループの都市数等のデータを格納する。

【0047】TSP処理部3は、記憶装置4に各都市毎のデータを格納し、並列計算機2からの応答を受信すると、前記都市データを更新しながらTSP処理を行う。この場合、TSP処理部3は、最初に記憶装置4からTSPの全都市座標データを読み込んだ際、各都市に前記のようなデータ構造を持たせ、全都市が出発都市と同一グループに属し、出発都市と到着都市が同じものと考えて、全都市座標から並列計算機2の各セルに処理を割り当てる。

【0048】並列計算機2は、TSP処理部3から与えられた問題から、与えられた全ての都市を巡回する巡回経路の内、最短経路を求める。全体の巡回経路は、都市データ構造の「ネクスト」で辿れるようにしておくので、経路が求められる度に、対象都市の「ネクスト」と「インバース」を書き換える。

【0049】また、都市を階層化する時に、同じセルに割り当てられる都市群はグループを作るが、それは各都市のグループという要素で記憶している。グループは、並列計算機2に分割される時に対象都市において書き換えを行う。最終的に全体の経路が求められた時には、全都市においてこのグループという要素はヌル (NULL) になっていることになる。

【0050】そして、最初に求められた巡回経路から全都市の経路が求められるまで、新たな部分問題を作成して、最初の全体問題と同様に部分経路を求める。求められた部分経路は全体の巡回経路の中に組み込む。

【0051】§4：部分問題作成時、及び部分問題処理後の都市データの説明・・・図5、図6参照
図5は部分問題作成時の都市データを示した図、図6は部分問題処理後の都市データを示した図である。

【0052】TSP処理時には、TSP処理部3は記憶装置4内の都市データを基に処理を行う。そして、最初の全体問題の処理を並列計算機2に依頼して処理した後、全都市を巡回した経路が求まっていない場合は、新たに部分問題を作成して並列計算機2に経路を求める処理を依頼する。

【0053】この場合、例えば、図5に示したように都市データが存在していたとする。この状態でTSP処理部3は、例えば、都市ID=5、10、7を含むグループを部分問題の都市データとして記憶装置4から読み出し、これらの都市データから部分問題を作成する。

【0054】そして、前記部分問題を並列計算機2で処

理させその結果のデータ（部分経路のデータ）を受信すると、TSP処理部3はこのデータを基に、記憶装置4内の都市データを更新する。この処理により図6に示したように、都市データの中身が変化して全体の都市巡回経路が詳細化していく。

【0055】§5：TSP処理と、階層化処理の説明・
・図7、図8参照

図7はTSP処理の説明図、図8は階層化処理説明図である。なお、図示のレベル1、レベル2は階層化のレベルを示している。また、●印は都市を示している。

【0056】前記のようにTSP処理を行う際、TSP処理部3は記憶装置4から必要なデータを読み込み、与えられた都市座標（ x 、 y ）から全都市を含む最小矩形を $m \times n$ に分割して、並列計算機2の $m \times n$ のセルにTSPの全都市をマッピングする。

【0057】そして、最初に解くべき全体問題（図示のレベル1参照）を作成し並列計算機2へ送信してTSP処理（与えられた全ての都市を巡回する経路の内、最短経路を求める処理）を依頼する。

【0058】前記問題を受信した並列計算機2は、最初は $m \times n$ のセル5を用いて都市を巡回する経路の内最短経路を求める処理を行う。そして、与えられた全都市を巡回する経路の内から最短経路が求められたら、その最短経路のデータをホストコンピュータ1のTSP処理部3へ転送して前記依頼に対し応答する。

【0059】TSP処理部3は、並列計算機2からの応答により経路データを受信すると、並列計算機2で求められた経路が全都市を巡回したものか否かを判断する。その結果、前記求められた経路が全都市を巡回するものでないと判断した場合、新たに部分問題を作成し、階層化を行うことにより（図示のレベル2参照）、再び並列計算機2へTSP処理を依頼する。そして、並列計算機2からの応答を受信すると、TSP処理部3は都市データの更新処理を行う。このようにして、並列計算機2のセル数が全都市数より少ない場合は、TSP処理を階層化して最適解に近いものを求める。

【0060】前記処理において、都市データ（city data）は x 、 y 座標（ x 、 y ）により、例えば、（0、10）、（3、5）、（2、12）・・・のように表されている。すなわち、全都市を含む x 、 y 座標に対し、座標軸上の各都市の存在する位置を（ x 、 y ）で表している。

【0061】従って、TSP処理部3が作成する問題データも、各都市ID0、1、2・・・に対し、0（0、0）、1（0、1）、2（0、2）・・・のように表されている。

【0062】ところで、初めの全都市の分割（全体問題）では、出発都市と到着都市が同じになる巡回経路を求めるが、階層化していく際の各部分問題では、出発都市と到着都市をすでに求められた経路から設定する。

【0063】この場合、TSP処理部3は、部分問題を作成する直前までに求められている巡回経路を基に、部分問題における代表都市の直前の訪問都市の位置が、部分問題の中のどの部分に割り当てられているかを調べ、そこから距離が一番近いセルに割り当てられている都市を部分問題の出発都市に設定する。

【0064】また、部分問題における代表都市の次の訪問都市の位置が部分問題の中のどの部分に割り当てられているかを調べ、そこから距離が一番近いセルに割り当てられている都市を到着都市に設定する。更に、部分問題を作成する際、出発都市、或いは到着都市に対応する位置に都市がない場合は、その位置に疑似的に作成した疑似都市を設定する。

【0065】前記のようにして各部分問題の解を得たTSP処理部3は、その経路中に疑似都市が含まれているか否かを判断し、疑似都市が含まれていた場合は、その疑似都市を除いた経路を求めて、その経路を全体の経路に組み込むと共に、疑似都市が含まれていない場合は、そのままの経路を全体の経路に組み込む処理を行う。

【0066】この場合、並列計算機2から部分問題の解が得られたら、その経路中に重複した経路が存在するか否かを判断し、重複した経路が存在した場合、到着都市の位置をずらして経路を修正する処理を行う。

【0067】このようにすることで、大規模な問題においても、それに対応する沢山の部分問題を作成し、各々を解いていくことで、全体の解を求めることができる。また、各部分問題は、最初に全体の巡回経路を求める解以外は、独立にその経路を探索することができる。そこで、1つの並列計算機2で、並列に複数の部分問題を探索させることができ、更に、解を求める時間を短縮することができる。

【0068】§6：部分問題の都市設定処理の説明・
・図9参照

図9は部分問題の都市設定処理説明図であり、A図は出発、到着、疑似都市の設定例を示した図、B図は経路重複修正例を示した図である。

【0069】部分問題を並列計算機2の各セルに再分割して割り付ける際に、その部分問題における出発都市と到着都市を決定する必要がある。これは、その部分問題を作成する直前までに求められている巡回経路から適当な都市を以下の手順により決定して設定する。

【0070】(1)：部分問題の代表都市のインバース（都市データにおける代表都市IDのインバース）が指している都市（直前の訪問都市）の位置が部分問題の中の何処の部分にマッピングされるかを調べ、そこから距離が一番近いセルに割り当てられている都市を出発都市に設定する。

【0071】(2)：前記(1)と同様にして部分問題の代表都市の「ネクスト」が指している都市（次の訪問都市）の座標が部分問題において、どこに位置にマッピン

グされるかを見て、そこから距離が一番近いセルに対応する都市を到着都市に設定する。

【0072】(3)：前記(1)、及び(2)において、それぞれ対応するセルに割り当てられる都市が存在しない場合は、疑似的に作成した疑似都市（実際には存在しない都市）を出発都市、或いは到着都市に設定する。

【0073】図9のA図には、4×4構成（セルが縦4個、横4個の構成）の並列計算機によるもので示しており、図の斜線部分は、都市が存在しているセルである。また、図には、直前の訪問都市（inverse city）、次の訪問都市（next city）、出発都市セル（start city cell）、疑似到着都市セル（dummy end city cell）が図示してある。

【0074】図示のように、出発都市と到着都市が割り当てられるセルは、その構成によりどこになるかは異なるが、長方形の4辺のいずれかに存在することになる。但し、疑似都市は、並列計算機の方から見た場合、実在の都市と何ら変わることはない、ただ単に部分問題の都市の数が増えるだけである。

【0075】前記のようにして出発都市、及び到着都市を設定すると、両者は同一になる場合と、そうでない場合が考えられる。同一の場合は、TSPを最初に解く場合と同様な経路を求めることになる。そうでない場合は、本来のTSPとは厳密には異なるが、部分問題の都市を巡る出発都市と到着都市が予め決まっている、非巡回経路を求めることになる。

【0076】例えば、図9のB図のように、前記出発都市と到着都市が同一になり、経路が重複する場合は、セルの位置を移動させて処理をやり直す。出発都市と到着都市が重複すると、内部で経路が循環したままとなることがあるので、解が求まらない状態となる。従って、到着都市に疑似都市を設定し出発都市と到着都市が重複しないようにして処理をやり直すことになる。

【0077】§7：都市巡回経路の詳細化と経路探索の階層化の説明・・・図10参照

図10は都市巡回経路の詳細化と経路探索の階層化の説明図である。TSP処理部3では、前記のようにして得られた部分経路を全体の巡回経路の中に組み込む。この時、疑似都市が入っている場合は、それを除いた形で組み込む。例えば、都市IDが「100」の疑似都市が入っていたとする。

【0078】この時、TSP処理部3は「100→2→5」を全体巡回経路「0→5」に組み込むと、巡回経路は「0→2→5」のようになる（各数字は都市ID、矢印は経路を示す）。

【0079】また、部分巡回経路の場合には、経路が重複してしまうと（図9のB図参照）全都市の解である巡回経路が長くなってしまふので、部分問題の解を求めた後に到着都市の位置をずらして処理をやり直している。

【0080】このようにして、全体の巡回経路を並列計

算機2からの部分経路を受け取る度に詳細化していくことで、最終的には、全体の出発都市から各都市のデータ構造の要素である「ネクスト」で繋がれた都市の巡回経路が得られることになる。

【0081】図10は、実際にTSP処理を行った結果の状態を示しており、解を求めていく過程で階層化が進められ、巡回経路が詳細化していく状態を示している。なお、この都市データでは、階層の深さは3（レベル1～レベル3）であり、部分問題の数は5である。また、解を求めるためにかかった時間は約1秒であった。

【0082】§8：都市巡回経路の解の説明・・・図11参照

図11は都市巡回経路の解を示した図であり、A図は都市の座標、B図は都市巡回経路の解を示した図である。

【0083】図11に示した例では、100の都市を対象とし、この100都市について都市巡回経路を実際に求めたものである。この場合、階層の深さは4であり、部分問題の数は34、全体の実行時間は13秒程度であった。距離はマンハッタン長で比較すると、シミュレーションで求めた略最適解より2割程度の誤差であった。

【0084】（他の実施例）以上実施例について説明したが、本発明は次のようにしても実施可能である。

(1)：疑似都市の設定方法は、前記実施例で説明した方法の外に、次のようにして設定しても良い。

【0085】例えば、図9において、メッシュ構造の4辺だけでなく、4角のセルにも割り当てるなどの特定のセルに限定する方法を採用しても良い。また、疑似都市を置かずに、実際に存在する都市の中で、部分問題の中で一番近い距離にある都市の割り当てられているセルに設定する方法を採用しても良い。

【0086】(2)：都市データの構造は、前記実施例で説明したものに限らず、前記実施例と同様なデータを含む任意のデータ構造で実施可能である。

(3)：都市データを格納しておく記憶装置は、磁気ディスク装置、光磁気ディスク装置、磁気テープ装置、半導体記憶装置等の任意の記憶装置で実施可能である。なお、TSP処理を行う際は、前記記憶装置から必要なデータを内部のワーク用メモリへ転送して処理を行い、処理が終了したら前記記憶装置へ格納しておく。

【0087】(4)：各部分問題（階層化レベル2以降の処理）は、独立してその経路を探索することができるので、1つの並列計算機で、並列に複数の部分問題を処理（部分問題の並列処理）させても良い。このようにすれば、更にTSPの解を求める時間を短縮することができる。

【0088】

【発明の効果】以上説明したように、本発明によれば次のような効果がある。

(1)：大規模な巡回セールスマン問題でも、階層化を取

り入れた解法により高速で解を求めることができる。また、メッシュ構造の並列計算機に分割した部分問題の処理を依頼することにより、解を求める時間を短縮することができる。

【0089】(2)：巡回セールスマン問題をメッシュ構造の並列計算機のセル数に応じて分割し、階層化して処理することにより、大規模の巡回セールスマン問題を高速に処理し、最適解に近いものをできるだけ早く求めることができる。また、部分問題の処理では、出発都市と到着都市を設定するが、所定の位置に前記各都市が存在しない場合は、疑似都市を設定することにより、処理が効率よく行える。

【0090】(3)：各部分問題（階層化レベル2以降の処理）は、独立してその経路を探索することができるので、1つの並列計算機で、並列に複数の部分問題を処理させれば、更にTSPの解を求める時間を短縮することができる。

【0091】前記の効果の外、各請求項に対応して次のような効果がある。

(4)：請求項1では、与えられた都市座標を基に全都市を含む最小矩形を並列計算機のセル数に応じて分割し、この分割した都市データを各セルに割り当てることで全体問題を作成し、その全体問題を並列計算機に処理させて解を求める全体問題処理手段と、並列計算機で求められた解の経路が全都市を巡るものでない場合、求められていない経路の都市群を含む最小矩形を、更に並列計算機のセル数に応じて分割することで階層化し、この分割して階層化した都市データを各セルに割り当てることで部分問題を作成し、その部分問題を並列計算機に処理させて解を求める部分問題処理手段を備えている。

【0092】従って、メッシュ構造の並列計算機を使用して、階層化した部分問題を並列処理で行うことにより、巡回セールスマン問題の解を高速に求められる。また、巡回セールスマン問題をメッシュ構造の並列計算機のセル数に応じて分割し、階層化して処理することにより、大規模の巡回セールスマン問題を高速に処理し、最適解に近いものをできるだけ早く求めることができる。

【0093】(5)：請求項2では、部分問題処理手段は、部分問題を作成する直前までに求められている巡回経路を基に、部分問題における代表都市の直前の訪問都市の位置が部分問題の中のどの部分に割り当てられているかを調べ、そこから距離が一番近いセルに割り当てられている都市を部分問題の出発都市に設定する出発都市設定手段を備えている。従って、部分問題を処理する際の出発都市の設定処理が確実にでき、TSP処理が効率よく行える。

【0094】(6)：請求項3では、部分問題処理手段は、部分問題を作成する直前までに求められている巡回経路を基に、部分問題における代表都市の次の訪問都市の位置が部分問題の中のどの部分に割り当てられている

かを調べ、そこから距離が一番近いセルに割り当てられている都市を到着都市に設定する到着都市設定手段を備えている。従って、部分問題を処理する際の到着都市の設定処理が確実にでき、TSP処理が効率よく行える。

【0095】(7)：請求項4では、部分問題処理手段は、部分問題を作成する際、出発都市、或いは到着都市に対応する位置に都市がない場合は、その位置に疑似的に作成した疑似都市を設定する疑似都市設定手段を備えている。

10 【0096】従って、出発都市や到着都市が所定の位置にない場合でも、疑似都市により出発都市、及び到着都市が確実に設定できるから、部分問題に対するTSP処理が効率よく行える。

【0097】(8)：請求項5では、並列計算機から部分問題の解が得られた場合、その経路中に疑似都市が含まれているか否かを判断し、疑似都市が含まれていた場合は、その疑似都市を除いた経路を求めて、その経路を全体の経路に組み込むと共に、疑似都市が含まれていない場合は、そのままの経路を全体の経路に組み込む経路組み込み処理手段を備えている。

20 【0098】従って、並列計算機からの部分経路を受け取る度に、全体の巡回経路を詳細化していくことで、全体の経路を順次確実に求められる。また、疑似都市を除いた経路を全体の経路に組み込むので、正確な経路が得られる。

30 【0099】(9)：請求項6では、並列計算機から部分問題の解が得られた場合、その経路中に重複した経路が存在するか否かを判断し、重複した経路が存在した場合、到着都市の位置をずらして経路を修正する経路重複修正手段を備えている。従って、常に正確な経路が求められる。

【図面の簡単な説明】

【図1】本発明の原理説明図である。

【図2】実施例のシステム構成図である。

【図3】実施例の処理フローチャートである。

【図4】実施例における都市データの説明図である。

40 【図5】実施例における部分問題作成時の都市データを示した図である。

【図6】実施例における部分問題処理後の都市データを示した図である。

【図7】実施例におけるTSP処理の説明図である。

【図8】実施例における階層化処理説明図である。

【図9】実施例における部分問題の都市設定処理説明図である。

【図10】実施例における都市巡回経路の詳細化と経路探索の階層化の説明図である。

【図11】実施例における都市巡回経路の解を示した図である。

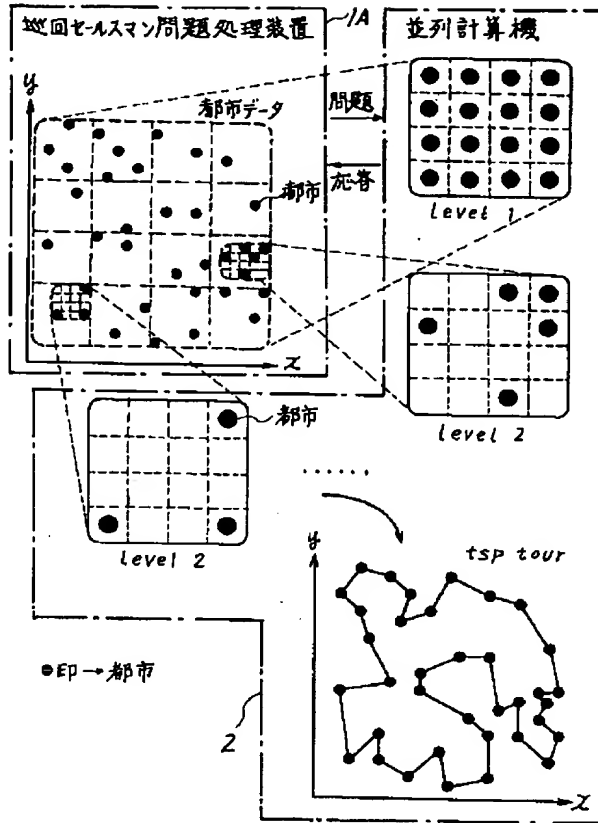
【符号の説明】

50 1 ホストコンピュータ

- 2 並列計算機
3 TSP処理部
4 記憶装置

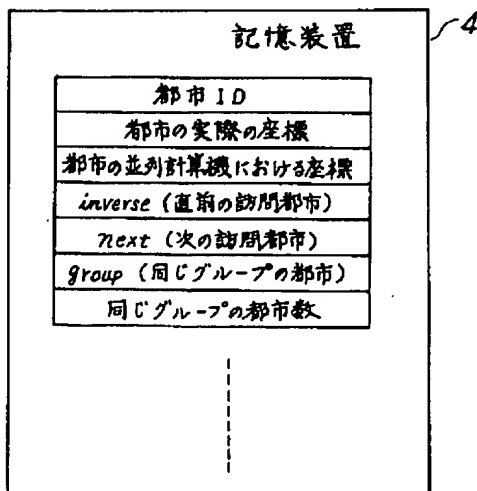
【図1】

本発明の原理説明図



【図4】

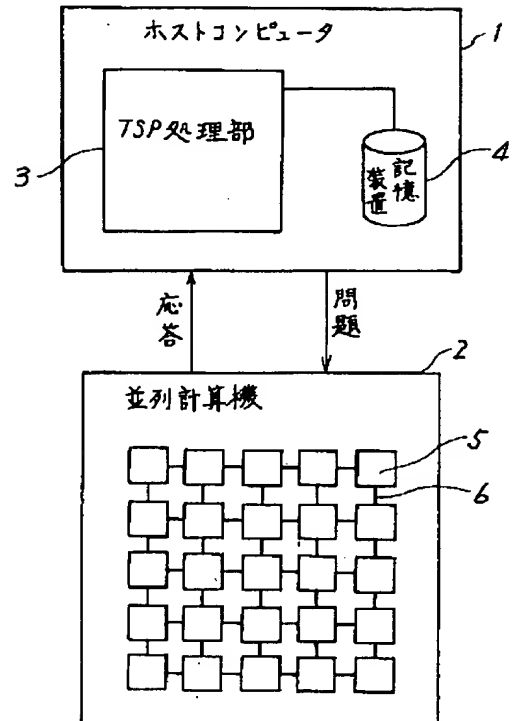
都市データの説明図



- 5 セル (プロセッサエレメント)
6 通信ネットワーク

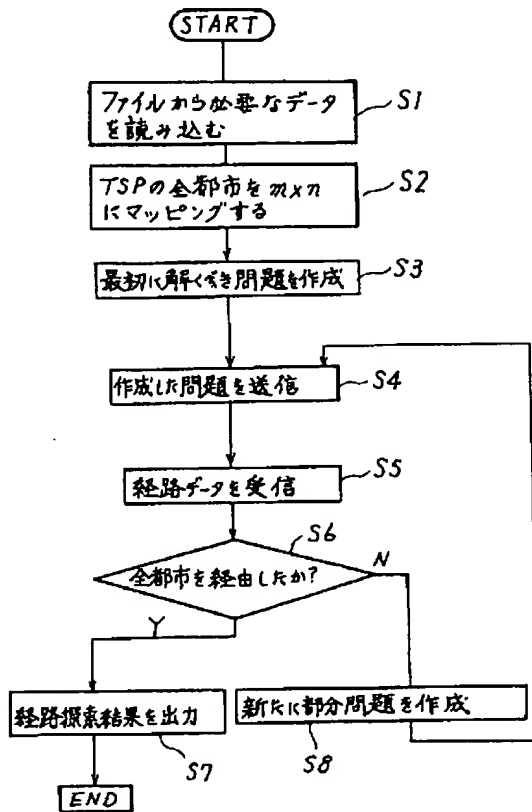
【図2】

実施例のシステム構成図



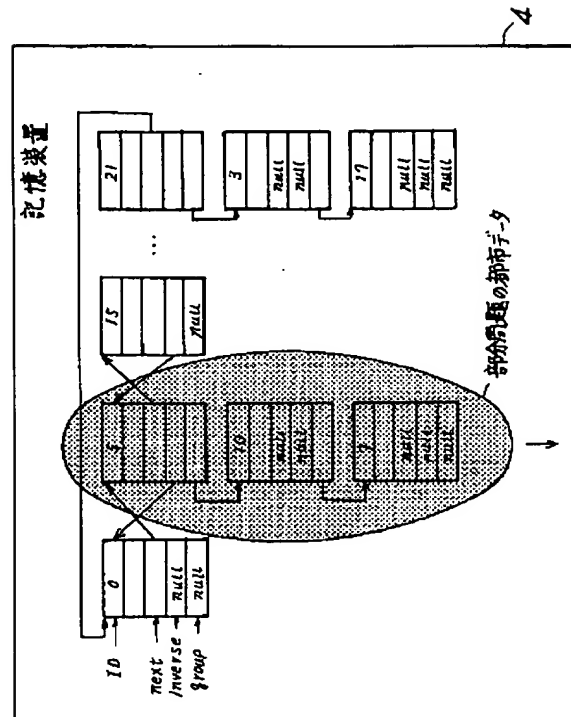
【図3】

実施例の処理フローチャート



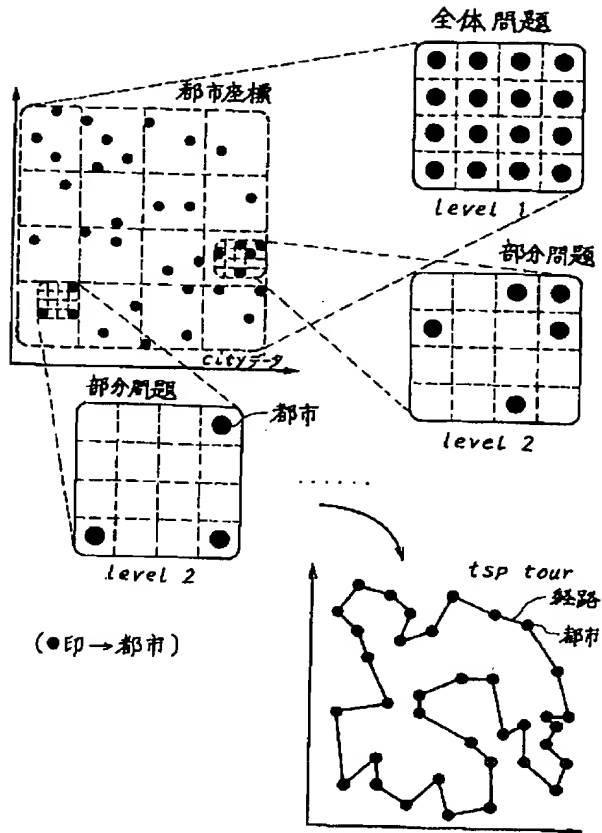
【図5】

部分問題作成時の都市データ



【図 8】

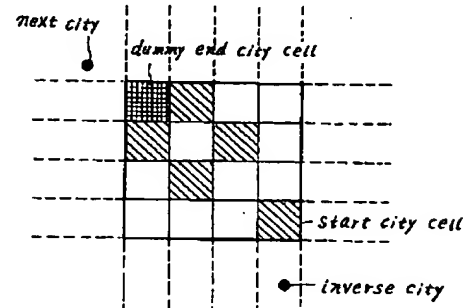
階層化処理説明図



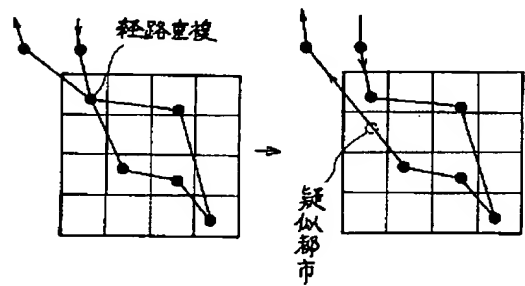
【図 9】

部分問題の都市設定処理説明図

A: 出発, 到着, 疑似都市の設定例

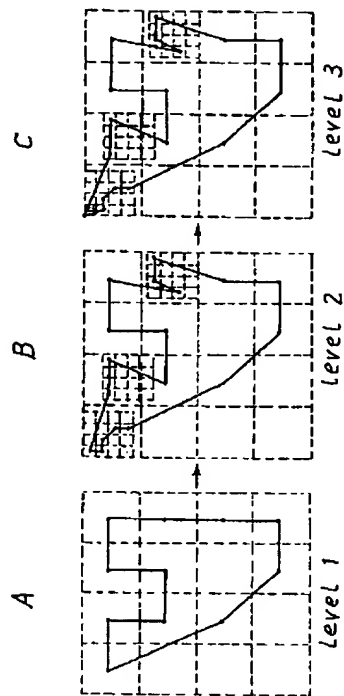


B: 経路重複修正例



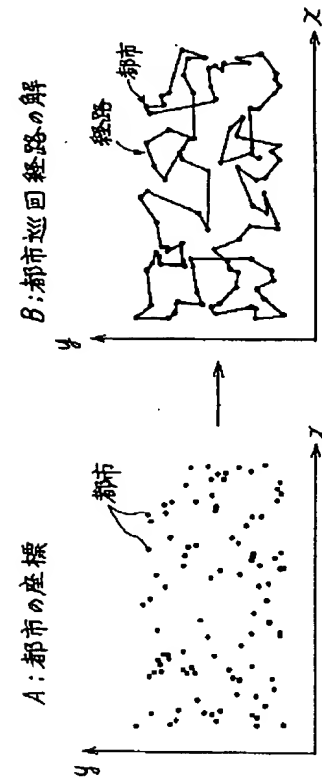
【図10】

都市巡回経路の詳細化と経路探索の階層化
の説明図



【図11】

都市巡回経路の解



フロントページの続き

(72)発明者 河村 薫
神奈川県川崎市中原区上小田中1015番地
富士通株式会社内

(72)発明者 稲田 由江
神奈川県川崎市中原区上小田中1015番地
富士通株式会社内
(72)発明者 三渡 秀樹
神奈川県川崎市中原区上小田中1015番地
富士通株式会社内